AMENDMENTS TO THE CLAIMS

Please cancel Claims 10-12 and 16, without prejudice.

Please amend Claims 1-9 and 13-15 as follows.

1. (Currently amended) A safety verification device of ~~a~~ an electronic reactive system such as a cipher communication system or control system for a nuclear reactor or aircraft, represented by a set of function symbols, a set of rewriting rules, a set of axioms, a set of terms, and a selected set of terms to be verified, said set of axioms being a set consisting only a commutative law and an associative law, and said safety verification device of a reactive system comprising~~:~~ a processing unit, a recording unit, a translation unit, a simulation unit and a set operation unit, wherein:

said set of function symbols, said set of rewriting rules, said set of axioms, said set of terms, and said selected set of terms to be verified are recorded in said recording unit;

~~a~~ said translation unit is controlled by said processing unit to read out said set of axioms and said set of terms from said recording unit and to generate ~~generating~~, under said set of axioms, a first equational tree automaton which accepts said set of terms;

~~a~~ said simulation unit is controlled by said processing unit to read out said set of rewriting rules, said set of axioms and said set of terms from said recording unit and to generate ~~generating~~, under said set of rewriting rules and said set of axioms and using said first equational tree automaton as initial data, a second equational tree automaton which accepts said set of terms and a set that comprises ~~comprising~~ terms derived from said set of terms; ~~and~~

~~a~~ said set operation unit is controlled by said processing unit ~~which~~ to generate ~~generates~~, using said second equational tree automaton and said selected set of terms to be verified, a fourth equational tree automaton by associating said second equational tree automaton with a third equational tree automaton which accepts said selected set of terms to be verified and to determine ~~determines~~ whether or not a set accepted by the fourth equational tree automaton is an empty set;

said second equational tree automaton is generated through first and second repetition processes;

wherein said first repetition process comprises:

(A) setting said first equational tree automaton to initial data;

(B) selecting an element p from a first group which consists of position information in a tree-structure when left sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element p is positioned at the end of said tree-structure;

(C) determining a set of terms by rewriting all terms which are included in a fifth equational tree automaton obtained in a last process performed according to the rewriting rule $f(c^{p.1}_{t1},\ldots,c^{p.n}_{tn})$ --> $c^{p}_{l|p}$, wherein a function symbol of said element p is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term $l_{|p}$ corresponding to said element p is described as $f(t_1,\ldots,t_n)$;

(D) obtaining a sixth equational tree automaton by performing repeatedly said (B) selecting and (C) determining processes regarding all elements p positioned at the ends of said tree-structure of said first group; and

wherein said second repeated process comprises:

(E) setting said sixth equational tree automaton to initial data;

(F) selecting an element q from a second group which consists of position information in a tree-structure when right sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element q is positioned at the end of said tree-structure;

(G) determining a set of terms by rewriting all terms which are included in a seventh equational tree automaton obtained in a last process performed according to the rewriting rule $f(d^{q.1}_{t1},\ldots,d^{q.n}_{tn})$ --> $d^{q}_{r|q}$, wherein a function symbol of said element q is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term $r_{|p}$ corresponding to said element q is described as $f(t_1,\ldots,t_n)$; and

(H) obtaining said second equational tree automaton by performing repeatedly said (F) selecting and (G) determining processes regarding all elements q positioned at the ends of said tree-structure of said second group.

2. (Currently amended) A safety verification device of ~~a~~ an electronic reactive system such as a cipher communication system or control system for a nuclear reactor or aircraft, represented by a set of function symbols, a set of rewriting rules, a set of axioms, a set of terms, and a selected term to be verified, said set of axioms being a set consisting only a commutative law and an associative law, and said safety verification device of a reactive system comprising~~:~~ a processing unit, a recording unit, a translation unit, a simulation unit and a set operation unit, wherein;

said set of function symbols, said set of rewriting rules, said set of axioms, said set of terms, and said selected term to be verified are recorded in said recording unit;

~~a~~ said translation unit is controlled by said processing unit to read out said set of axioms and said set of terms from said recording unit and to generate ~~generating~~, under said set of axioms, a first equational tree automaton which accepts said set of terms;

~~a~~ said simulation unit is controlled by said processing unit to read out said set of rewriting rules, said set of axioms and said set of terms from said recording unit and to generate ~~generating~~, under said set of rewriting rules and said set of axioms and using said first equational tree automaton as initial data, a second equational tree automaton which accepts said set of terms and a set that comprises ~~comprising~~ terms derived from said set of terms; ~~and~~

~~a~~ said set operation unit is controlled by said processing unit to determine ~~determining~~ whether or not said second equational tree automaton accepts said selected term to be verified;

said second equational tree automaton is generated through first and second repetition processes;

wherein said first repetition process comprises:

(A) setting said first equational tree automaton to initial data;

(B) selecting an element p from a first group which consists of position information in a tree-structure when left sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element p is positioned at the end of said tree-structure;

(C) determining a set of terms by rewriting all terms which are included in a fifth equational tree automaton obtained in a last process performed according to the rewriting rule

$f(c^{p.1}_{t1},\ldots,c^{p.n}_{tn}) \text{-->} c^{p}_{l|p}$, wherein a function symbol of said element p is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term $l_{|p}$ corresponding to said element p is described as $f(t_1,\ldots,t_n)$;

(D) obtaining a sixth equational tree automaton by performing repeatedly said (B) selecting and (C) determining processes regarding all elements p positioned at the ends of said tree-structure of said first group; and

wherein said second repeated process comprises:

(E) setting said sixth equational tree automaton to initial data;

(F) selecting an element q from a second group which consists of position information in a tree-structure when right sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element q is positioned at the end of said tree-structure;

(G) determining a set of terms by rewriting all terms which are included in a seventh equational tree automaton obtained in a last process performed according to the rewriting rule $f(d^{q.1}_{t1},\ldots,d^{q.n}_{tn}) \text{-->} d^{q}_{r|q}$, wherein a function symbol of said element q is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term $r_{|p}$ corresponding to said element q is described as $f(t_1,\ldots,t_n)$; and

(H) obtaining said second equational tree automaton by performing repeatedly said (F) selecting and (G) determining processes regarding all elements q positioned at the ends of said tree-structure of said second group.

3. (Currently amended) A safety verification device of a reactive system according to claim 1, wherein said set of function symbols is a set comprising function symbols representing encryption, decryption and communication processing as elements,

said set of rewriting rules is a set comprising as an element a rule representing that encrypted information is returned to plaintext by decryption,

said selected term to be verified is confidential information, and said set of terms is a set of knowledge of each of subjects that exchange confidential information, and a set of knowledge of a subject that monitors the information exchanged between said subjects.

4. (Currently amended) A safety verification method of ~~a~~ an electronic reactive system such as a cipher communication system or control system for a nuclear reactor or aircraft, represented by a set of function symbols, a set of rewriting rules, a set of axioms, a set of terms, and a selected set of terms to be verified, said set of axioms being a set consisting only a commutative law and an associative law, said method being executed by a computer comprising a processing unit and a recording unit, and said method comprising:

a first step in which said processing unit reads out said set of axioms and said set of terms from said recording unit and ~~of~~ generates ~~generating~~, under said set of axioms, a first equational tree automaton which accepts said set of terms;

a second step in which said processing unit reads out said set of rewriting rules, said set of axioms and said set of terms from said recording unit and ~~of~~ generates ~~generating~~, under said set of rewriting rules and said set of axioms and using said first equational tree automaton as initial data, a second equational tree automaton which accepts said set of terms and a set ~~of~~ that comprises terms derived from said set of terms; and

a third step in which said processing unit ~~of~~ generates ~~generating~~, using said second equational tree automaton and said selected set of terms to be verified, a fourth equational tree automaton by associating said second equational tree automaton with a third equational tree automaton which accepts said selected set of terms to be verified and said processing unit determines ~~determining~~ whether or not a set accepted by the fourth equational tree automaton is an empty set, wherein said second step comprises first and second repetition processes;

wherein said first repetition process comprises:

(A) setting said first equational tree automaton to initial data;

(B) selecting an element p from a first group which consists of position information in a tree-structure when left sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element p is positioned at the end of said tree-structure;

(C) determining a set of terms by rewriting all terms which are included in a fifth equational tree automaton obtained in a last process performed according to the rewriting rule $f(c^{p.1}{}_{t1},\ldots,c^{p.n}{}_{tn}) \dashrightarrow c^{p}{}_{l|p}$, wherein a function symbol of said element p is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term $l_{|p}$ corresponding to said element p is described as $f(t_1,\ldots,t_n)$;

(D) obtaining a sixth equational tree automaton by performing repeatedly said (B) selecting and (C) determining processes regarding all elements p positioned at the ends of said tree-structure of said first group; and

wherein said second repeated process comprises:

(E) setting said sixth equational tree automaton to initial data;

(F) selecting an element q from a second group which consists of position information in a tree-structure when right sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element q is positioned at the end of said tree-structure;

(G) determining a set of terms by rewriting all terms which are included in a seventh equational tree automaton obtained in a last process performed according to the rewriting rule $f(d^{q.1}{}_{t1},\ldots,d^{q.n}{}_{tn}) \dashrightarrow d^{q}{}_{r|q}$, wherein a function symbol of said element q is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term $r_{|p}$ corresponding to said element q is described as $f(t_1,\ldots,t_n)$; and

(H) obtaining said second equational tree automaton by performing repeatedly said (F) selecting and (G) determining processes regarding all elements q positioned at the ends of said tree-structure of said second group.

5. (Currently amended) A safety verification method of ~~a~~ an electronic reactive system such as a cipher communication system or control system for a nuclear reactor or aircraft, represented by a set of function symbols, a set of rewriting rules, a set of axioms, a set of terms, and a selected term to be verified, said set of axioms being a set consisting only a commutative

law and an associative law, said method being executed by a computer comprising a processing unit and a recording unit, and said method comprising:

a first step in which said processing unit reads out said set of axioms and said set of terms from said recording unit and ~~of~~ generates ~~generating~~, under said set of axioms, a first equational tree automaton which accepts said set of terms;

a second step in which said processing unit reads out said set of rewriting rules, said set of axioms and said set of terms from said recording unit and ~~of~~ generates ~~generating~~, under said set of rewriting rules and said set of axioms and using said first equational tree automaton as initial data, a second equational tree automaton which accepts said set of terms and a set ~~of~~ that comprises terms derived from said set of terms; and

a third step in which said processing unit ~~of~~ determines ~~determining~~ whether or not said second equational tree automaton accepts said selected term to be verified, wherein said second step comprises first and second repetition processes;

wherein said first repetition process comprises:

(A) setting said first equational tree automaton to initial data;

(B) selecting an element p from a first group which consists of position information in a tree-structure when left sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element p is positioned at the end of said tree-structure;

(C) determining a set of terms by rewriting all terms which are included in a fifth equational tree automaton obtained in a last process performed according to the rewriting rule $f(c^{p.1}_{t_1},\ldots,c^{p.n}_{t_n})$ --> $c^{p}_{l|p}$, wherein a function symbol of said element p is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term $l_{|p}$ corresponding to said element p is described as $f(t_1,\ldots,t_n)$;

(D) obtaining a sixth equational tree automaton by performing repeatedly said (B) selecting and (C) determining processes regarding all elements p positioned at the ends of said tree-structure of said first group; and

wherein said second repeated process comprises:

(E) setting said sixth equational tree automaton to initial data;

(F) selecting an element q from a second group which consists of position information in a tree-structure when right sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element q is positioned at the end of said tree-structure;

(G) determining a set of terms by rewriting all terms which are included in a seventh equational tree automaton obtained in a last process performed according to the rewriting rule $f(d^{q.1}_{t1},\ldots,d^{q.n}_{tn})$ --> $d^{q}_{r|q}$, wherein a function symbol of said element q is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term $r_{|p}$ corresponding to said element q is described as $f(t_1,\ldots,t_n)$; and

(H) obtaining said second equational tree automaton by performing repeatedly said (F) selecting and (G) determining processes regarding all elements q positioned at the ends of said tree-structure of said second group.

6. (Currently amended) A safety verification method of a reactive system according to claim 4, wherein said set of function symbols is a set comprising function symbols representing encryption, decryption and communication processing as elements,

said set of rewriting rules is a set comprising as an element a rule representing that encrypted information is returned to plaintext by decryption,

said selected term to be verified is confidential information, and

said set of terms is a set of knowledge of each of subjects that exchange confidential information, and a set of knowledge of a subject that monitors the information exchanged between said subjects.

7. (Currently amended) A computer-readable recording medium containing a reactive system safety verification computer program, said reactive system being an electronic system such as a cipher communication system or control system for a nuclear reactor or aircraft, said computer program being executed by a computer comprising a processing unit and a recording unit, and said computer program comprising:

a first program code which makes said processing unit to accept ~~accepts~~ an input of a procedure represented by a set of function symbols, a set of rewriting rules, a set of axioms, a set of terms, and a selected set of terms to be verified and to record said procedure in said recording unit;

a second program code which makes said processing unit to read out said set of axioms and said set of terms from said recording unit and to generate ~~generates~~, under said set of axioms consisting only of a commutative law and an associative law, a first equational tree automaton which accepts said set of terms;

a third program code which makes said processing unit to read out said set of rewriting rules, said set of axioms and said set of terms from said recording unit and to generate ~~generates~~, under said set of rewriting rules and said set of axioms and using said first equational tree automaton as initial data, a second equational tree automaton which accepts said set of terms and a set ~~of~~ that comprises terms derived from said set of terms; and

a fourth program code which makes said processing unit to generate ~~generates~~, using said second equational tree automaton and said selected set of terms to be verified, a fourth equational tree automaton by associating said second equational tree automaton with a third equational tree automaton which accepts said selected set of terms to be verified and to determine ~~determines~~ whether or not a set accepted by the fourth equational tree automaton is an empty set, wherein said second program code makes said processing unit to execute first and second repetition processes;

wherein said first repetition process comprises:

(A) setting said first equational tree automaton to initial data;

(B) selecting an element p from a first group which consists of position information in a tree-structure when left sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element p is positioned at the end of said tree-structure;

(C) determining a set of terms by rewriting all terms which are included in a fifth equational tree automaton obtained in a last process performed according to the rewriting rule $f(c^{p.1}_{t1},\ldots,c^{p.n}_{tn})$ --> $c^{p}_{t|p}$, wherein a function symbol of said element p is described as f, argument terms are described as $t_1,...,t_n$, and a term l_{lp} corresponding to said element p is described as $f(t_1,...,t_n)$;

(D) obtaining a sixth equational tree automaton by performing repeatedly said (B) selecting and (C) determining processes regarding all elements p positioned at the ends of said tree-structure of said first group; and

wherein said second repeated process comprises:

(E) setting said sixth equational tree automaton to initial data;

(F) selecting an element q from a second group which consists of position information in a tree-structure when right sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element q is positioned at the end of said tree-structure;

(G) determining a set of terms by rewriting all terms which are included in a seventh equational tree automaton obtained in a last process performed according to the rewriting rule $f(d^{q.1}_{t1},...,d^{q.n}_{tn})$ --> d^{q}_{rlq}, wherein a function symbol of said element q is described as f, argument terms are described as $t_1,...,t_n$, and a term r_{lp} corresponding to said element q is described as $f(t_1,...,t_n)$; and

(H) obtaining said second equational tree automaton by performing repeatedly said (F) selecting and (G) determining processes regarding all elements q positioned at the ends of said tree-structure of said second group.

8. (Currently amended) A computer-readable recording medium containing a reactive system safety verification computer program, said reactive system being an electronic system such as a cipher communication system or control system for a nuclear reactor or aircraft, said computer program being executed by a computer comprising a processing unit and a recording unit, and said computer program comprising:

a first program code which makes said processing unit to accept ~~accepts~~ an input of a procedure represented by a set of function symbols, a set of rewriting rules, a set of axioms, a set of terms, and a selected term to be verified and to record said procedure in said recording unit;

a second program code which makes said processing unit to read out said set of axioms and said set of terms from said recording unit and to generate ~~generates~~, under said set of axioms consisting only of a commutative law and an associative law, a first equational tree automaton which accepts said set of terms;

a third program code which makes said processing unit to read out said set of rewriting rules, said set of axioms and said set of terms from said recording unit and to generate ~~generates~~, under said set of rewriting rules and said set of axioms and using said first equational tree automaton as initial data, a second equational tree automaton which accepts said set of terms and a set ~~of~~ that comprises terms derived from said set of terms; and

a fourth program code which makes said processing unit to determine ~~determines~~ whether or not said second equational tree automaton accepts said selected term to be verified, wherein said second program code makes said processing unit to execute first and second repetition processes;

wherein said first repetition process comprises:

(A) setting said first equational tree automaton to initial data;

(B) selecting an element p from a first group which consists of position information in a tree-structure when left sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element p is positioned at the end of said tree-structure;

(C) determining a set of terms by rewriting all terms which are included in a fifth equational tree automaton obtained in a last process performed according to the rewriting rule $f(c^{p.1}_{t1},...,c^{p.n}_{tn})$ --> $c^{p}_{t|p}$, wherein a function symbol of said element p is described as f, argument terms are described as $t_1,...,t_n$, and a term $l_{|p}$ corresponding to said element p is described as $f(t_1,...,t_n)$;

(D) obtaining a sixth equational tree automaton by performing repeatedly said (B) selecting and (C) determining processes regarding all elements p positioned at the ends of said tree-structure of said first group; and

wherein said second repeated process comprises:

(E) setting said sixth equational tree automaton to initial data;

(F) selecting an element q from a second group which consists of position information in a tree-structure when right sides of equations, each of said equations corresponding to a rewriting rule in said set of rewriting rules, are described in tree-structure, wherein said element q is positioned at the end of said tree-structure;

(G) determining a set of terms by rewriting all terms which are included in a seventh equational tree automaton obtained in a last process performed according to the rewriting rule $f(d^{q.1}_{t1},\ldots,d^{q.n}_{tn})$ --> d^{q}_{rlq}, wherein a function symbol of said element q is described as f, argument terms are described as $t_1,\ldots,t_n$, and a term r_{lp} corresponding to said element q is described as $f(t_1,\ldots,t_n)$; and

(H) obtaining said second equational tree automaton by performing repeatedly said (F) selecting and (G) determining processes regarding all elements q positioned at the ends of said tree-structure of said second group.

9. (Currently amended) A computer-readable recording medium containing a reactive system safety verification computer program according to claim 7, wherein said set of function symbols is a set comprising function symbols representing encryption, decryption and communication processing as elements,

said set of rewriting rules is a set comprising as an element a rule representing that encrypted information is returned to plaintext by decryption,

said selected term to be verified is confidential information, and

said set of terms is a set of knowledge of each of subjects that exchange confidential information, and a set of knowledge of a subject that monitors the information exchanged between said subjects.

10-12. (Cancelled)

13. (Currently amended) A safety verification device of a reactive system according to claim 2, wherein said set of function symbols is a set comprising function symbols representing encryption, decryption and communication processing as elements,

said set of rewriting rules is a set comprising as an element a rule representing that encrypted information is returned to plaintext by decryption,

said selected term to be verified is confidential information, and

said set of terms is a set of knowledge of each of subjects that exchange confidential information, and a set of knowledge of a subject that monitors the information exchanged between said subjects.

14. (Currently amended) A safety verification method of a reactive system according to claim 5, wherein said set of function symbols is a set comprising function symbols representing encryption, decryption and communication processing as elements,

said set of rewriting rules is a set comprising as an element a rule representing that encrypted information is returned to plaintext by decryption,

said selected term to be verified is confidential information, and

said set of terms is a set of knowledge of each of subjects that exchange confidential information, and a set of knowledge of a subject that monitors the information exchanged between said subjects.

15. (Currently amended) A computer-readable recording medium containing a reactive system safety verification computer program according to claim 8, wherein said set of function symbols is a set comprising function symbols representing encryption, decryption and communication processing as elements,

said set of rewriting rules is a set comprising as an element a rule representing that encrypted information is returned to plaintext by decryption,

said selected term to be verified is confidential information, and

said set of terms is a set of knowledge of each of subjects that exchange confidential information, and a set of knowledge of a subject that monitors the information exchanged between said subjects.

16. (Cancelled)